CU-HTK April 2002 Switchboard System

Phil Woodland, Gunnar Evermann, Mark Gales, Thomas Hain,
Andrew Liu, Gareth Moore, Dan Povey & Lan Wang

May 7th 2002

Cambridge University Engineering Department

Rich Transcription Workshop 2002

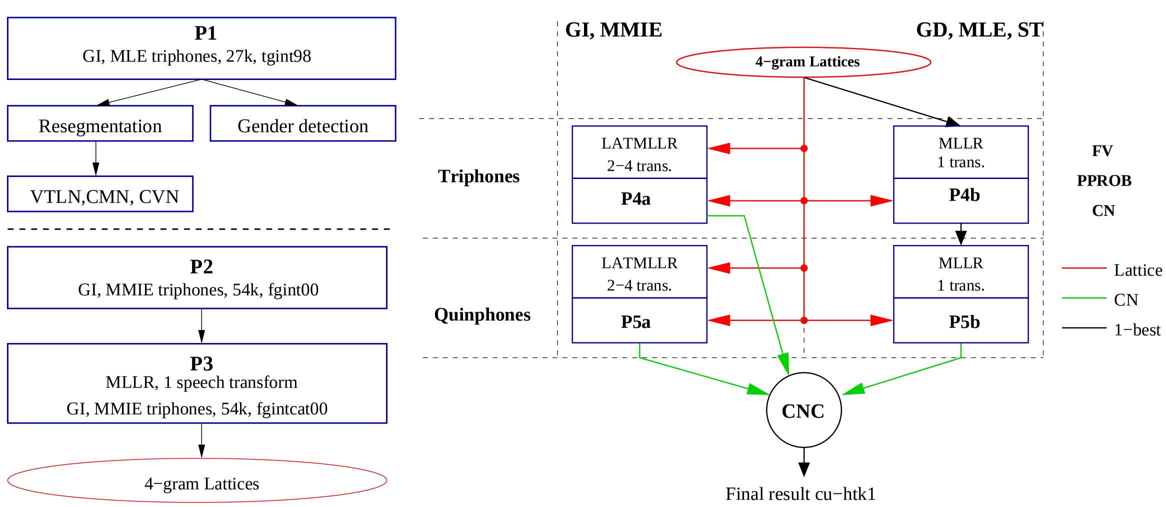
Overview

- Review of CU-HTK 2001 system
- Minimum Phone Error (MPE) training
- HLDA
- Speaker Adaptive Training
- Single Pronunciation dictionaries
- 2002 system & results
- Fast contrast systems
- Conclusions

Review of CU-HTK 2001 System: Basic Features

- Front-end
  - Reduced bandwidth 125–3800 Hz
  - 12 MF-PLP cepstral parameters + C0 and 1st/2nd derivatives
  - Side-based cepstral mean and variance normalisation
  - Vocal tract length normalisation in training and test
- Decision tree state clustered, context dependent triphone & quinphone models: MMIE and MLE versions
- Generate lattices with MLLR-adapted models
- Rescore using iterative lattice MLLR + Full-Variance transform adaptation
- Posterior probability decoding via confusion networks
- System combination

2001 System Structure



Acoustic Training/Test Data

h5train00 248 hours Switchboard (Swbd1), 17 hours CallHome English (CHE)

h5train00sub 60 hours Swbd1, 8 hours CHE

h5train02 h5train00 + LDC cell1 corpus (without dev01/eval01 sides) extra 17 hours of data

Development test sets

dev01 40 sides Swbd2 (eval98), 40 sides Swbd1 (eval00), 38 sides Swbd2 cellular (dev01-cell)

dev01sub half of the **dev01** selected to give similar WER to full set

eval98 40 sides Swbd2 (eval98-swbd2), 40 sides of CHE (eval98-che)

2001 System Results on dev01 set

		Swbd1	Swbd2	Cellular	Total
P1	VTLN/gender det	31.7	46.9	48.1	42.1
P2	initial trans.	23.5	38.6	39.2	33.7
P3	lat gen	21.1	36.0	36.7	31.2
P4a	MMIE tri	20.0	33.5	34.0	29.1
P4b	MLE tri	21.3	35.0	35.4	30.5
P5a	MMIE quin	19.8	33.2	33.4	28.7
P5b	MLE quin	20.2	34.0	34.2	29.4
CNC	P5a+P4a+P5b	18.3	31.9	32.1	27.3

%WER on dev01 for all stages of 2001 system

- final confidence scores have NCE 0.254

Minimum Phone Error & Other Discriminative Criteria

- MMIE maximises the posterior probability of the correct sentence
  Problem: sensitive to outliers
- MCE maximises a smoothed approximation to the sentence accuracy
  Problem: cannot easily be implemented with lattices; scales poorly to long sentences
- Criterion we evaluate in testing is word error rate: makes sense to maximise something similar to it
- MPE uses smoothed approximation to phone error but can use lattice-based implementation developed for MMIE
- Note that MPE is an approximation to phone error *in a word recognition context* i.e. uses word-level recognition, but scoring is on a phone error basis.
- Can directly maximise a smoothed *word* error rate → Minimum Word Error (MWE). Performance for MWE slightly worse than MPE, so main focus here on MPE

MPE Objective Function

- Maximise the following function:

$$\mathcal{F}_{\mathrm{MPE}}(\lambda) = \sum_{r}^{R} \frac{\sum_s p_\lambda(\mathcal{O}_r|s)^\kappa P(s)\mathrm{RawAccuracy}(s)}{\sum_s p_\lambda(\mathcal{O}_r|s)^\kappa P(s)}$$

  where λ are the HMM parameters, $\mathcal{O}_r$ the speech data for file r, κ a probability scale and $P(s)$ the LM probability of s
- $\mathrm{RawAccuracy}(s)$ measures the number of phones correctly transcribed in sentence s (derived from *word* recognition).
  i.e. # correct phones in s − # inserted phones in s
- $\mathcal{F}_{\mathrm{MPE}}(\lambda)$ is weighted average of $\mathrm{RawAccuracy}(s)$ over all s
- Scale acoustic log-likelihoods by scale κ.
- Criterion is to be maximised, not minimised (for compatibility with MMIE)

Lattice Implementation of MMIE: Review

- Generate lattices marked with time information at HMM level
  - Numerator (num) from correct transcription
  - Denominator (den) for confusable hypotheses from recognition
- Use Extended Baum-Welch (Gopalakrishnan et al, Normandin) updates e.g. for means

$$\hat{\mu}_{jm} = \frac{\left\{\theta_{jm}^{\mathrm{num}}(\mathcal{O}) - \theta_{jm}^{\mathrm{den}}(\mathcal{O})\right\} + D\mu_{jm}}{\left\{\gamma_{jm}^{\mathrm{num}} - \gamma_{jm}^{\mathrm{den}}\right\} + D}$$

  - Gaussian occupancies (summed over time) are γ_{jm} from forward-backward
  - $\theta_{jm}(\mathcal{O})$ is sum of data, weighted by occupancy.
- For rapid convergence use Gaussian-specific D-constant
- For better generalisation broaden posterior probability distribution
  - Acoustic scaling
  - Weakened language model (unigram)

Lattice Implementation of MPE

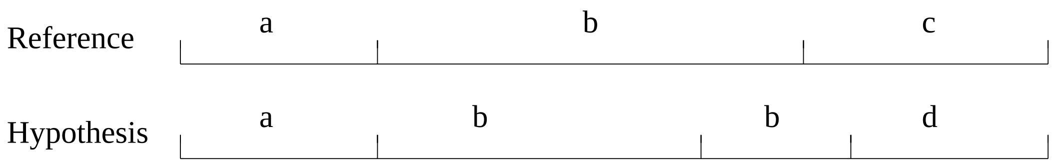
- Problem: $\mathrm{RawAccuracy}(s)$, defined on sentence level as (#correct - #inserted) requires alignment with correct transcription
- Express $\mathrm{RawAccuracy}(s)$ as a sum of $\mathrm{PhoneAcc}(q)$ for all phones q in the sentence hypothesis s:

$$\mathrm{PhoneAcc}(q) = \left\{ \begin{array}{l} 1 \text{ if correct phone} \\ 0 \text{ if substitution} \\ -1 \text{ if insertion} \end{array} \right\}$$

- Calculating $\mathrm{PhoneAcc}(q)$ still requires alignment to reference transcription
- Use an approximation to $\mathrm{PhoneAcc}(q)$ based on time-alignment information
  - compute the proportion e that each hypothesis phone overlaps the reference
  - gives a lower-bound on true value of $\mathrm{RawAccuracy}(s)$

Approximating PhoneAcc using Time Information

$$\text{PhoneAcc}(q) = \left\{ \begin{array}{l} -1 + 2e \text{ if same phone} \\ -1 + e \text{ if different phone} \end{array} \right\}$$



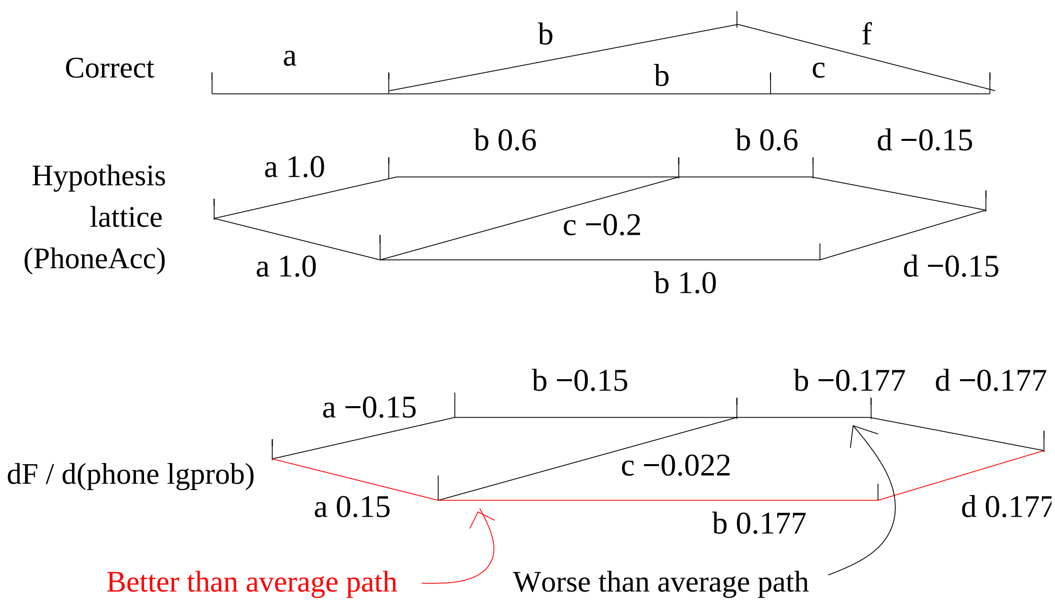
Proportion e	1.0	0.8	0.2	0.15	0.85
−1 + (correct:2*e, incorrect:e)	1.0	0.6	−0.6	−0.85	−0.15
Max of above	1.0	0.6	−0.6		−0.15

Approximated sentence raw accuracy from above = 0.85

Exact value of raw accuracy: 2 corr − 1 ins = 1

PhoneAcc Approximation For Lattices

Calc $\text{PhoneAcc}(q)$ for each phone q, then find $\frac{\partial \mathcal{F}_{\text{MPE}}(\lambda)}{\partial \log p(q)}$ (forward-backward)



Applying Extended Baum-Welch to MPE

- Use EBW update formulae as for MMIE but with modified MPE statistics
- For MMIE, the occupation probability for an arc q equals $\frac{1}{\kappa}\frac{\partial \mathcal{F}_{\mathrm{MMIE}}(\lambda)}{\partial \log p(q)}$ for numerator ($\times -1$ for the denominator). The denominator occupancy-weighted statistics are subtracted from the numerator in the update formulae
- Statistics for MPE update use $\frac{1}{\kappa}\frac{\partial \mathcal{F}_{\mathrm{MPE}}(\lambda)}{\partial \log p(q)}$ of the criterion w.r.t. the phone arc log likelihood which can be calculated efficiently
- Either MPE numerator or denominator statistics are updated depending on the sign of $\frac{\partial \mathcal{F}_{\mathrm{MPE}}(\lambda)}{\partial \log p(q)}$, which is the "MPE arc occupancy"
- After accumulating statistics, apply EBW equations
- EBW is viewed as a gradient descent technique and can be shown to be a valid update for MPE.

Improved Generalisation using I-smoothing

- Use of discriminative criteria can easily cause over-training
- Get smoothed estimates of parameters by combining Maximum Likelihood (ML) and MPE objective functions for each Gaussian
- Rather than globally interpolate (H-criterion), amount of ML depends on the occupancy for each Gaussian
- I-smoothing adds τ samples of the average ML statistics for each Gaussian. Typically $\tau = 50$.
  - For MMIE scale numerator counts appropriately
  - For MPE need ML counts in addition to other MPE statistics
- I-smoothing essential for MPE (& helps a little for MMIE)

MPE Training Results (I)

	Train	eval98	eval98 change
MLE	41.8	46.6	–
MMIE	30.1	44.3	-2.3
MMIE (τ=200)	32.2	43.8	-2.8
MPE (τ=50)	27.9	43.1	-3.5

%WER for h5train00sub HMMs (68h train). Train uses lattice unigram LM

	Train	eval98	eval98 change
MLE baseline	47.2	45.6	–
MMIE	37.7	41.8	-3.8
MMIE (τ=200)	35.8	41.4	-4.2
MPE (τ=100)	34.4	40.8	-4.8

%WER for h5train00 HMMs (265h train). Train uses lattice unigram LM

- I-smoothing reduces the error rate with MMIE by 0.3-0.4% abs
- MPE/I-smoothing gives around 1% abs lower WER than previous MMIE results

MPE Training Results (II)

	Train	eval98	eval98 change
MLE	41.8	46.6	–
MPE ($\tau = 0$)	28.5	50.7	+4.1
MPE ($\tau = 25$)	27.9	43.1	-3.5
MWE ($\tau = 25$)	25.9	43.3	-3.3

%WER for h5train00sub HMMs (68h train). Train uses lattice unigram LM

- Training set WER reduces with/without I-smoothing
- I-smoothing essential for test-set gains with MPE
- Minimum Word Error (MWE) better than MPE on train
- MWE generalises less well than MPE

MPE Summary

- Introduced MPE (& MWE) to give error-rate based discriminative training
  - Less affected by outliers than MMIE training
  - Smoothed approximation to phone error in word recognition system
  - Approximate reference-hypothesis alignment
  - Use same lattice-based training framework developed for MMIE
  - Compute suitable MPE statistics so still use Extended Baum-Welch update
  - Use I-smoothing to improve generalisation (essential for MPE)
- MPE/I-smoothing reduces WER over previous MMIE approach by 1% abs
- MPE/I-smoothing improvements over MLE essentially constant when applied to HMM sets with more mixture components up to 28
- MPE/I-smoothing used for all triphone and quinphone model sets in CU-HTK April 2002 Switchboard evaluation system

New cellular training data

- Extended training set by adding cell1 data to form h5train02
- Removed cellular data appearing in dev01 and eval01: 17.4 hours remain

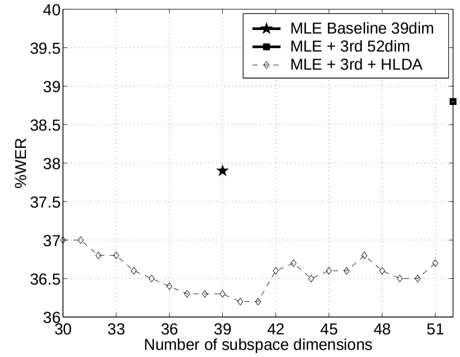
	Swbd1	Swbd2	Cellular	Total
h5train00	25.2	42.1	42.5	36.5
h5train02	24.9	41.3	41.7	35.8
h5train02 weighted	24.9	41.0	41.4	35.7

%WER on dev01sub using 16-mix MLE triphones with 2001 fgintcat lattices

- Improvements for cellular and non-cellular!
- After adaptation typically WER reduced by 0.5% abs overall
- Helps robustness of HLDA estimation

Heteroscedastic Linear Discriminant Analysis (HLDA)

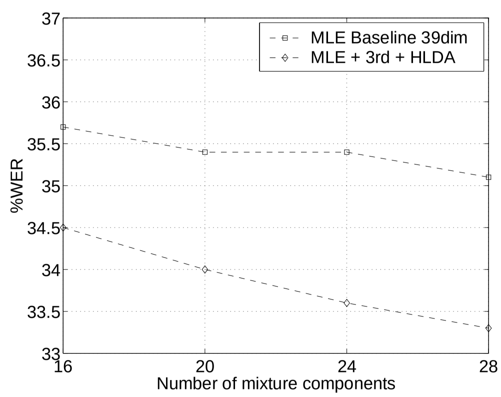
- Maps feature space to lower dimensional globally decorrelated [Kumar 1997]
- allows using higher order cepstral differentials up to 3rd order (52 dimensional) [Matsoukas et al. 2001]
- Transform estimation is through EM algorithm in an iterative fashion
  - using Fisher-ratio values to select nuisance dimensions
  - modelling nuisance dimensions by a global Gaussian
  - diagonal covariance constraint



%WER on dev01sub, 2001 fgintcat lattices, h5train00sub

HLDA: Triphone Results

- Triphone h5train02 systems rescoring 2001 fgintcat lattices on dev01sub



-	non-HLDA	HLDA
MLE training	35.1%	33.3%
MPE training	31.4%	30.1%
MPE + Lattice MLLR	28.9%	27.5%

%WER on dev01sub using 28mix h5train02 triphones, 2001 fgintcat lattices

- Mixture Splitting more beneficial with HLDA
- Gains still present after MPE and adaptation

Speaker Adaptive Training (SAT)

- The objective of SAT is to remove inter-speaker variability in training data, which should lead to more "compact" speaker independent models [Anastasakos 1996]
- Constrained MLLR is used to generate a single full-matrix transform for each side which is then applied to the feature space during training [Gales 1997]
- The re-estimation of model parameters for SAT uses either conventional ML or discriminative criterion (MMIE or MPE).
- Starting with the normal speaker independent model, four iterations of interleaved transform estimation and model parameter updating are performed to obtain ML-SAT models.
- Six iterations of MPE training are used to get MPE-SAT models. Transforms are not updated (ML-SAT transforms).

SAT: Triphone Results

- Results on dev01sub with 1-best unconstrained global MLLR adaptation

	#Iteration	Swbd1	Swbd2	Cellular	Total
ML		20.2	35.8	36.4	30.7
MPE	8	18.0	33.6	34.3	28.5
ML-SAT	4	19.2	35.0	35.2	29.7
MPE-SAT	2	18.0	33.4	34.0	28.4
MPE-SAT	4	18.0	33.2	33.6	28.1
MPE-SAT	6	17.6	33.0	33.6	28.0

%WER on dev01sub using 28mix HLDA triphones trained on h5train02, 2001 fgintcat lattices

- SAT reduces effectiveness of MPE, but increases convergence speed

Single Pronunciation Dictionaries (SPron)

60% of pronunciation variants in dictionaries only describe phoneme substitutions which can be implicitly modelled by Gaussian mixtures.

- Systematically remove all pronunciation variants
  Based on frequency in alignment of the training data.

- If words were observed in the training data:
  - Merging of variants with phoneme substitutions
  - Only most frequent variant is kept

- For words not observed:
  - Merging of variants with phoneme substitutions
  - Deletion of variants predicted to be less frequent
  - Random deletion

SPron Results

	train	Swbd1	Swbd2	Cellular	Total
MPron	MLE	21.5	37.9	38.1	32.4
SPron	MLE	21.3	37.7	37.4	32.0
MPron	MPE	19.1	35.0	35.6	29.8
SPron	MPE	19.6	34.9	34.9	29.7

%WER on dev01sub using 28-mix triphone models (h5train02), HLDA and pprobs, 2001 fgintcat lattices

- SPron models show lower word error rates on more difficult data

- Similar results were obtained with quinphones

	Swbd1	Swbd2	Cellular	Total
MPron	16.8	31.7	32.1	26.8
SPron	17.0	31.5	31.7	26.7
MPron + SPron	16.4	31.0	31.0	26.1

%WER on dev01sub using 28-mix triphone models (h5train02), HLDA, pprobs, LatMLLR, CN, 2001 fgintcat lattices

- Difference of system outputs: 0.6% WER from 2-fold system combination

Dictionary and Language Models

Dictionary:

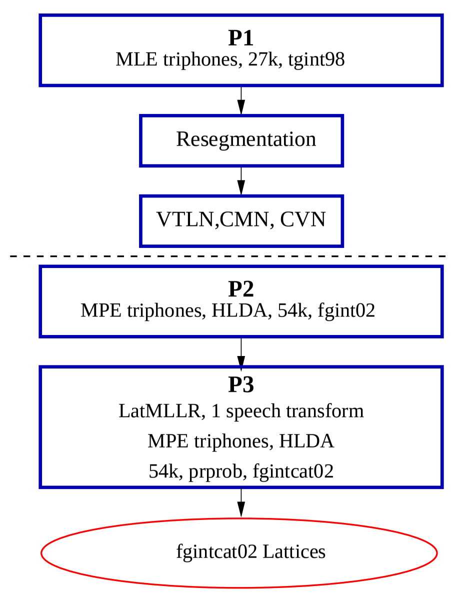
- 54598 words: Hub5 vocabulary (incl. cell1) plus top 50k words of Broadcast News data (0.38% OOV on eval98 and 0.17% on dev01cellular)
- Multiple pronunciation dictionary (based on LIMSI'93 + TTS). Probabilities estimated from forced alignment

Language models

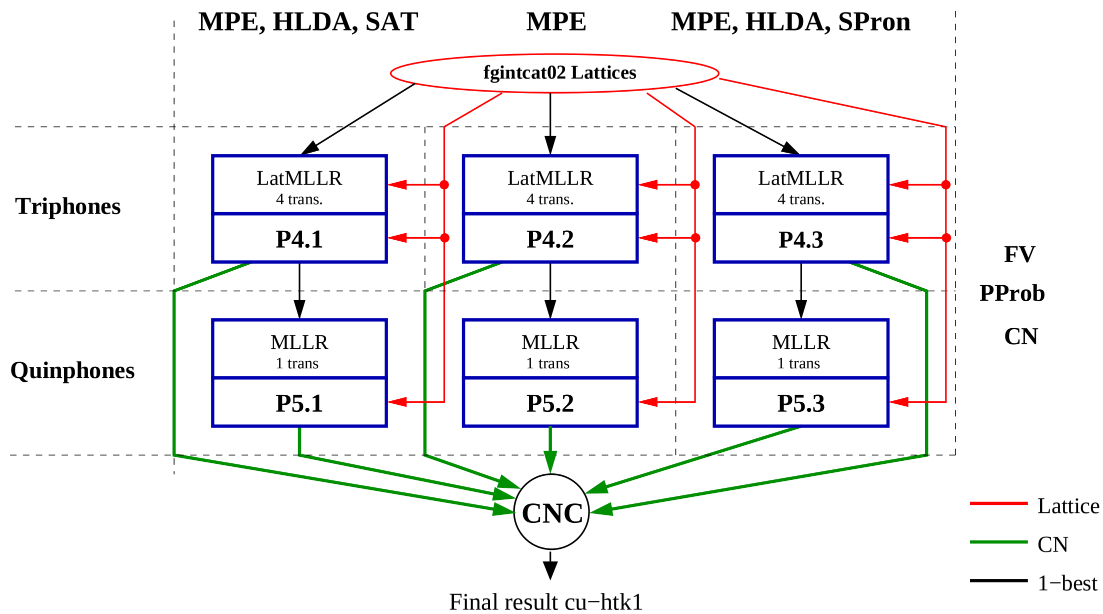
- Training data
  - 204MW Broadcast News
  - 3MW 1998 Hub5 + 3MW 2000 MSU Hub5 + 0.2MW cell1
- 3-fold interpolated/merged bigram, trigram, and 4-gram word LMs
- Class based trigram model (350 classes) to smooth word LM
- Hub5 LMs use modified Kneser-Ney discounting with SRILM toolkit. Broadcast News + class LMs trained using HTK LM toolkit

2002 System - Lattice Generation

- Stages similar to previous years
- What is different?
  - MPE triphone models
  - More mixture components (28 mix)
  - HLDA
  - Lattice MLLR based on P2 output
  - Use of pronunciation probabilities
  - New language models
  - Use of HDecode



2002 system – Rescoring & Combination



Results on dev01 set

		Swbd1	Swbd2	Cellular	Total
P1	trans for VTLN	31.7	46.9	48.1	42.1
P2	trans for MLLR	20.1	34.7	34.3	29.6
P3	lat gen	18.5	32.2	31.1	27.2
P4.1	SAT tri	17.5	30.7	29.6	25.9
P4.2	non-HLDA tri	18.8	31.4	31.0	27.0
P4.3	SPron tri	18.0	31.0	29.7	26.2
P5.1	SAT quin	17.2	30.8	29.2	25.7
P5.2	non-HLDA quin	18.5	31.8	30.6	26.9
P5.3	SPron quin	18.1	31.1	28.8	25.9
CNC	P4.[123]+P5.[123]	16.4	29.2	27.4	24.2

%WER on dev01 for all stages of 2002 system

- final confidence scores have NCE 0.238

Results on eval02 set

		Swbd1	Swbd2	Cellular	Total
P1	trans for VTLN	35.6	44.6	50.5	44.0
P2	trans for MLLR	24.6	30.9	34.8	30.4
P3	lat gen	22.5	28.0	31.3	27.5
P4.1	SAT tri	21.6	26.3	29.6	26.1
P4.2	non-HLDA tri	22.3	27.4	31.2	27.2
P4.3	SPron tri	21.5	26.6	29.1	26.0
P5.1	SAT quin	21.5	25.5	28.6	25.4
P5.2	non-HLDA quin	22.4	26.7	30.7	26.9
P5.3	SPron quin	21.5	26.4	28.8	25.8
CNC	P4.[123]+P5.[123]	19.8	24.3	27.0	23.9

%WER on eval02 for all stages of 2002 system

- final confidence scores have NCE 0.289

CU-HTK over the years on dev01 set

- Fast simple single model system (cu-htk2 contrast) 70xRT

year	Swbd1	Swbd2	Cellular	Total
2000	22.1	36.2	37.0	31.7
2001	20.6	34.8	35.6	30.2
2002	17.7	31.4	30.5	26.4

- Full multi-model eval system (cu-htk1) 300xRT

year	Swbd1	Swbd2	Cellular	Total
2000	19.3	32.5	33.2	28.3
2001	18.3	31.9	32.1	27.3
2002	16.4	29.2	27.4	24.2

Computation for 2002 cu-htk1 system

Pass	Speed ($\times$RT)
P1	12
P2	11
P3	37
P4.[123]	31
P5.[123]	147

Times based on Pentium III 1GHz

- Adaptation for P3 (lattice MLLR) 6xRT
- Model marked lattices for P4 (3 sets) 48xRT
- Lattice MLLR/FV estimation (3 sets) 19xRT
- 1-best MLLR/FV (3 quinphone sets) 9xRT

Total: 320xRT

Faster Contrast Systems

- Later stages in the full system only provide small, incremental benefits at high costs. Run only first stages as a contrast:

cu-htk2 Generate confusion networks from P3 rescoring lattices, i.e. only VTLN HLDA MPE Triphones, no rescoring, no quinphones. 67xRT

cu-htk3 Combine three triphone systems (P4.[123]). 165xRT

Results on eval02

	xRT	Swbd1	Swbd2	Cellular	Total	NCE
cu-htk1	320	19.8	24.3	27.0	23.9	0.289
cu-htk2	67	21.8	27.1	30.2	26.7	0.305
cu-htk3	165	20.5	25.3	28.0	24.8	0.288

%WER on eval02 of 2002 primary and contrast systems

10xRT System

- Based on initial stages of the full cu-htk1 system with tighter pruning and modified architecture
- Uses fast decoders employed in CUHTK-Entropic 1998 Hub4 10xRT system and HDecode
- Stages:
  - P1 (initial transcription) eval98 MLE triphones, trigram LM
  - VTLN, least squares linear regression adaptation
  - P2 (lattice generation) HLDA VTLN MPE triphones, tgint02 LM
  - Lattice expansion with fgint02 LM
  - MLLR adaptation (2 speech + 1 silence transform)
  - P3 (lattice rescoring): eval02 HLDA VTLN MPE triphones
  - Confusion networks for decoding + confidence scores

10xRT System: Results

- Results on dev01

system	xRT	Swbd1	Swbd2	Cellular	Total
2001 cu-htk1	300	18.3	31.9	32.1	27.3
2002 cu-htk1	320	16.4	29.2	27.4	24.2
2002 cu-htk4	10	18.3	31.9	31.0	27.0

- Results on eval02

	Swbd1	Swbd2	Cellular	Total
P1	36.7	46.3	51.3	45.2
P2tg	24.1	29.5	33.3	29.3
+ fg	23.4	28.9	32.3	28.5
P3	23.2	28.3	31.5	27.9
P3-cn	22.3	27.7	31.0	27.2

10xRT System: Computation

Run times on eval02

Pass		Speed ($\times$RT)
P1	coding	0.008
	initial trans.	1.300
	alignment	0.041
	VTLN	0.296
P2	adaptation	0.156
	lat gen	5.085
	lat expansion	0.098
P3	adaptation	0.477
	lat rescoring	1.735
	confnet	0.025
Total		9.221

Times based on Athlon 1900+ (1.6GHz), Redhat Linux, Intel C Compiler

Conclusions

- Improvements over 2001 Hub5 CU-HTK system come from
  - MPE/I-smoothing training (1%)
  - HLDA and 3rd differentials (1.5%)
  - More mixture components: 28 or 24 vs 16 (1%)
  - New cellular data (0.5%)
  - Revised LM (0.2%)
  - SAT combined with MPE
  - SPron dictionary
  - HDecode produces improved lattices
- Overall absolute reduction in WER over 2001:
  - 3.1% from full system
  - 3.8% from cu-htk2 triphone only, no system combination
- First 10xRT HTK Switchboard system
  - Fast version of cu-htk2
  - Only 0.5% abs worse than cu-htk2 on eval02
  - Lower word error on dev01 than 2001 full system

HTK3 Development

- Available for free download from `http://htk.eng.cam.ac.uk` since Sep 2000
- More than 12000 registered users and active mailing lists
- Gradually more features of the internal CU-HTK are incorporated in HTK3
- As part of DARPA EARS project CUED will develop HTK3 further:
  - Integrate LM tools for training of large word/class-based n-grams
  - Implement lattice processing tools
  - Make available HTK-based LVR decoder HDecode (used for P3 and P4)
  - Incorporate discriminative training tools
  - Provide infrastructure for standard tasks/testsets (e.g. recipes, simple models and lattices for past WSJ/BN/Switchboard evals).
- ICASSP'02 HTK meeting: Tue 14.May 6pm. "Palani Sailfish" meeting room, Renaissance, Orlando